

12 April 1987

UNIVERSITI SAINS MALAYSIA

Peperiksaan Semester Kedua

Sidang 1986/87

CST202 - Kejuruteraan Sofwer

Tarikh: 14 April 1987

Masa: 9.00 pagi - 12.00 t/hari
(3 Jam)

Sila pastikan bahawa:

Kertas peperiksaan ini mengandungi lima soalan. Jawab mana-mana EMPAT.

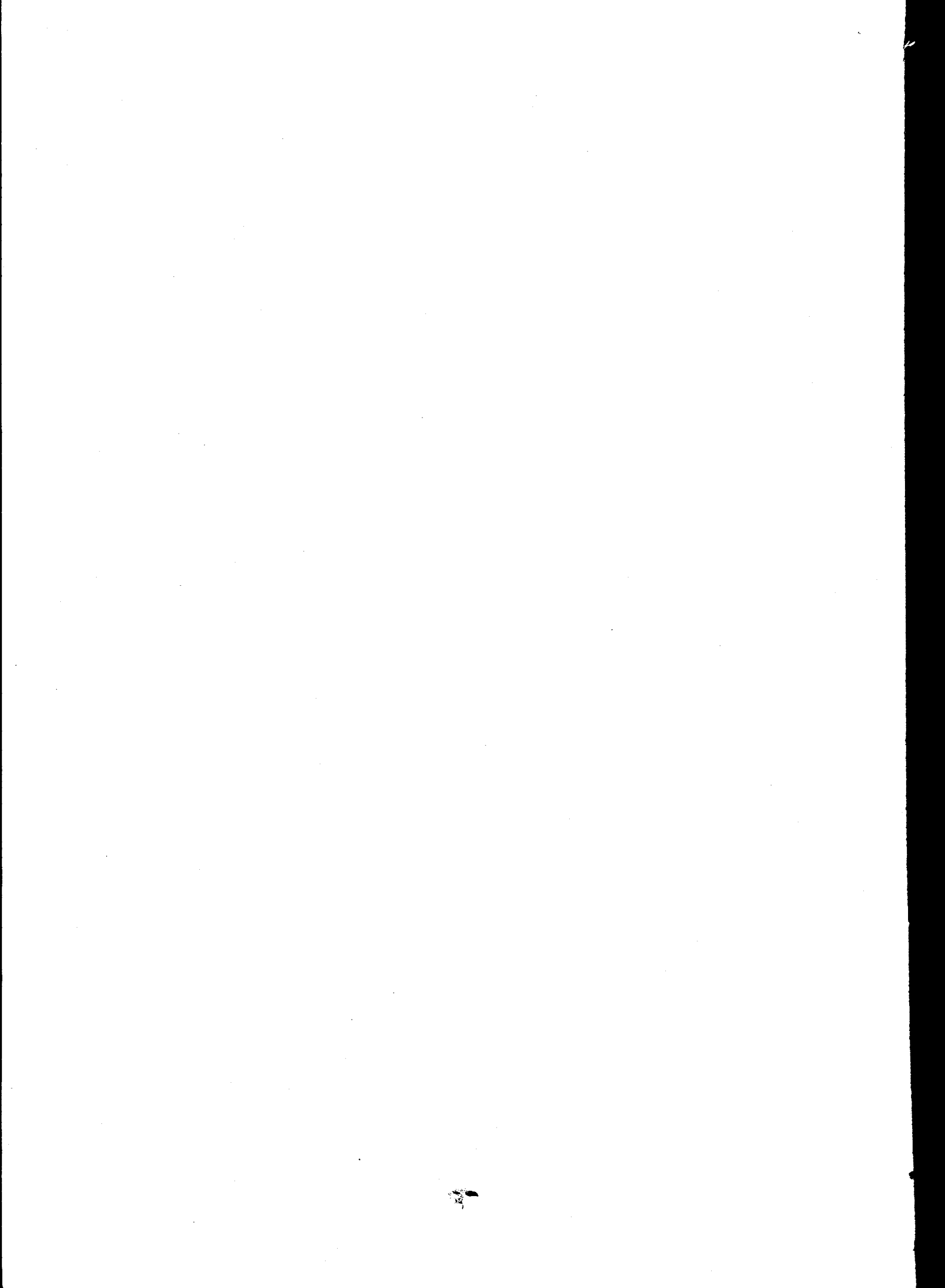
Setiap soalan mengandungi nilai yang sama.

Semua soalan MESTI dijawab dalam Bahasa Malaysia.

Bahagian A

1. (a) Terdapat 3 peringkat utama di dalam proses kejuruteraan sofwer. Nyatakan 3 peringkat tersebut serta beri huraian ringkas mengenai apa yang berlaku di setiap peringkat.
- (b) Nyatakan kepentingan dokumentasi sesuatu program/sofwer.
- (c) Huraikan perbezaan di antara ujian kotak-hitam (black-box testing) dan ujian kotak-putih (white-box testing) pada amnya dan juga dalam konteks sofwer.

(100/100)



2. (a) (i) Sebelum kita dapat membuktikan ketepatan sesuatu fungsi, kita perlu membuat satu andaian mengenai fungsi itu. Nyatakan andaian tersebut.

- (ii) Apabila kita membuktikan hubungan

$$\{P\} \quad S \quad \{Q\}$$

di mana S ialah badan fungsi, satu syarat untuk S mesti dipenuhi agar petua bukti fungsi

$$\frac{\{P\} \quad S \quad \{Q\}}{\forall x (P \Rightarrow Q_{f(x)}^f)}$$

tidak membawa pada percanggahan. Nyatakan syarat tersebut.

Tunjukkan satu contoh di mana syarat tersebut tidak dipenuhi dan bagaimana pembuktiannya membawa pada percanggahan.

- (b) Berikut ialah petua bukti untuk membuktikan ketepatan tatacara tanpa pembolehubah sejagat.

$$\frac{\{P\} \quad S \quad \{Q\}}{\forall x(\{P\}p(x) \{Q\})}$$

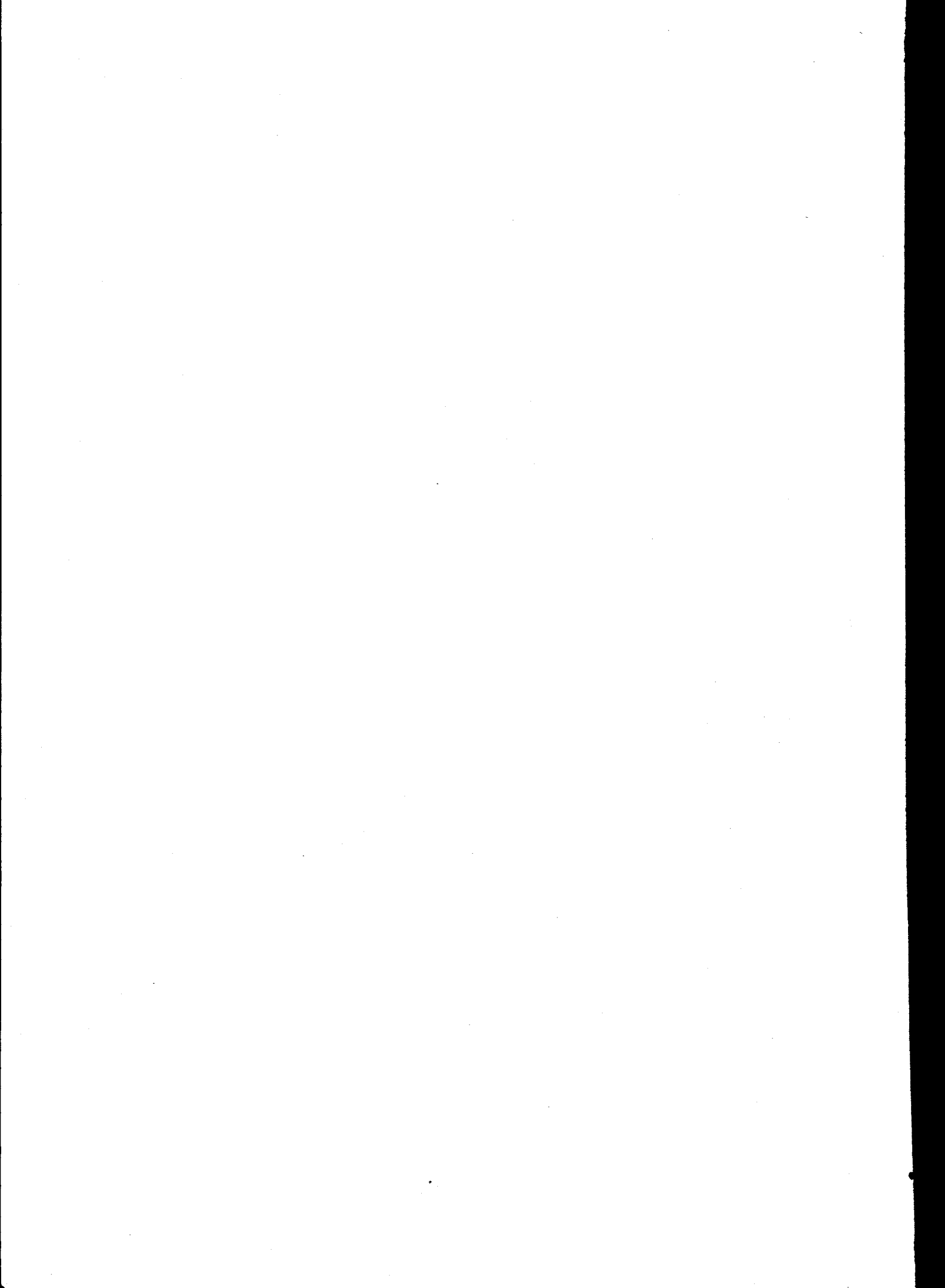
$$\frac{\forall x(\{P\} p(x) \{Q\})}{\{P_a^x\}p(a) \{Q_a^x\}}$$

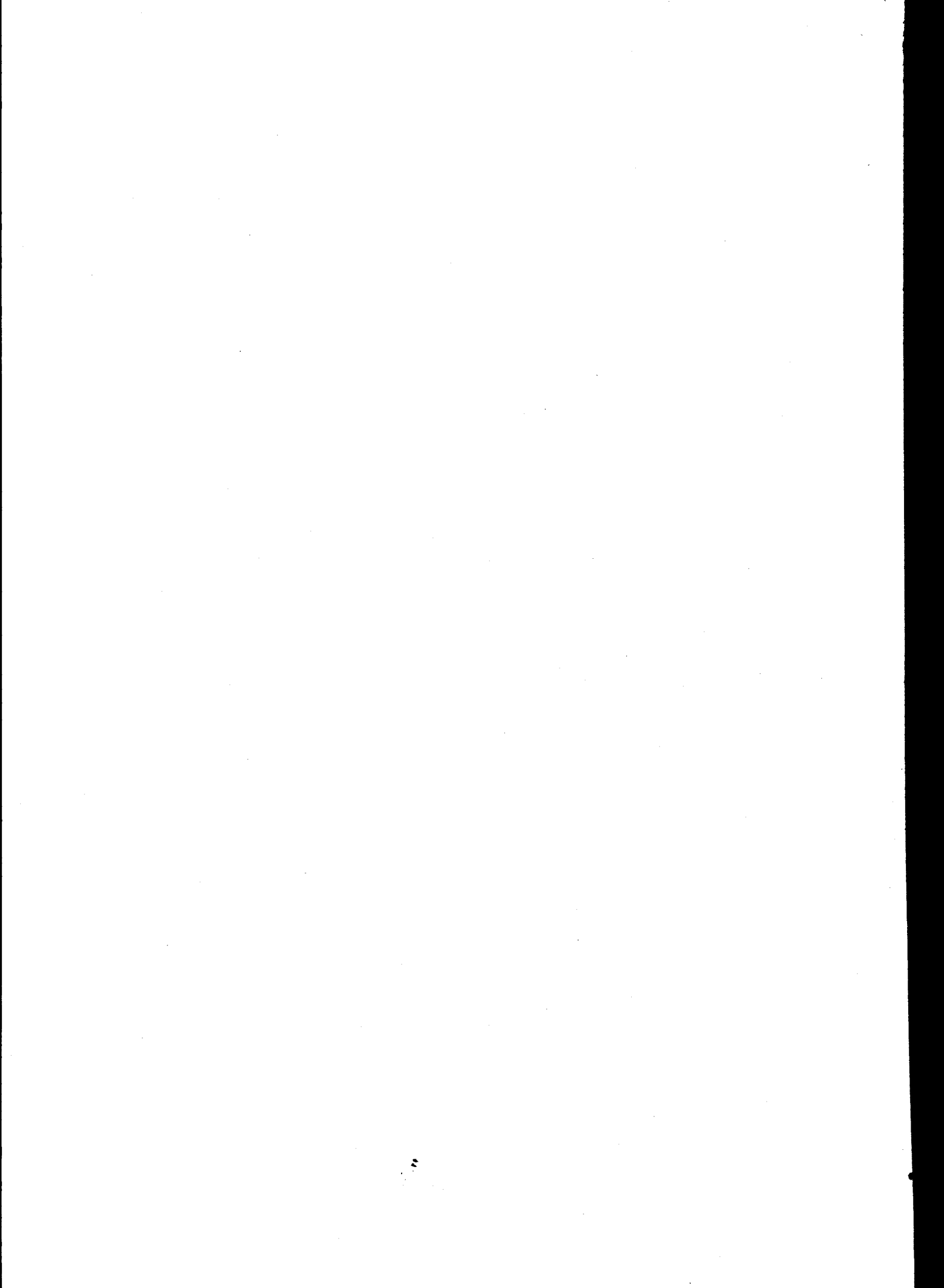
Di mana S ialah badan tatacara p dan x ialah senarai parameter formal.

Ada satu syarat atau kondisi yang tidak dinyatakan oleh petua itu, di mana tanpa syarat tersebut petua bukti yang diberi akan membawa pada percanggahan.

Nyatakan syarat atau kondisi tersebut. Tunjukkan satu contoh di mana syarat tersebut tidak dipenuhi dan bagaimana pembuktiannya membawa pada percanggahan.

(100/100)





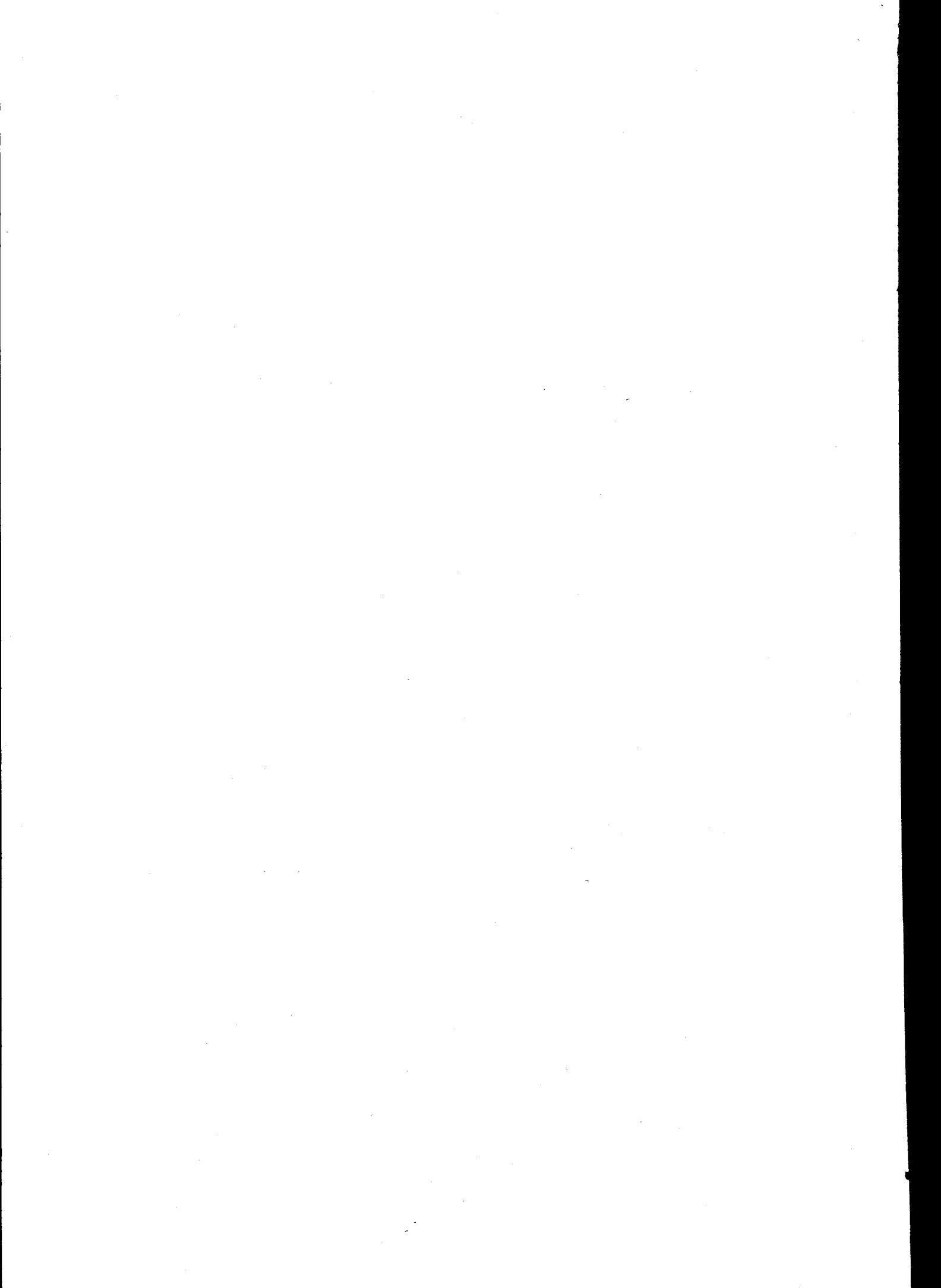
(i) Apakah kesan panggilan $p(a[i])$ ke atas susunan a dan pembolehubah x jika cara penggantian bagi parameter x dalam tatacara p ialah melalui

(1) pembolehubah

(2) nilai

(ii) Apakah perbezaan di antara penggantian melalui pembolehubah dan penggantian melalui nilai?

(100/100)



Bahagian B

4. (a) Terangkan setiap yang berikut:

(i) Kaedah rekabentuk atas-bawah

(ii) Hubungan kesetaraan

(20/100)

(b) Buktikan ketepatan seluruh bagi program yang berikut:
(Andaikan bahawa a dan b adalah integer)

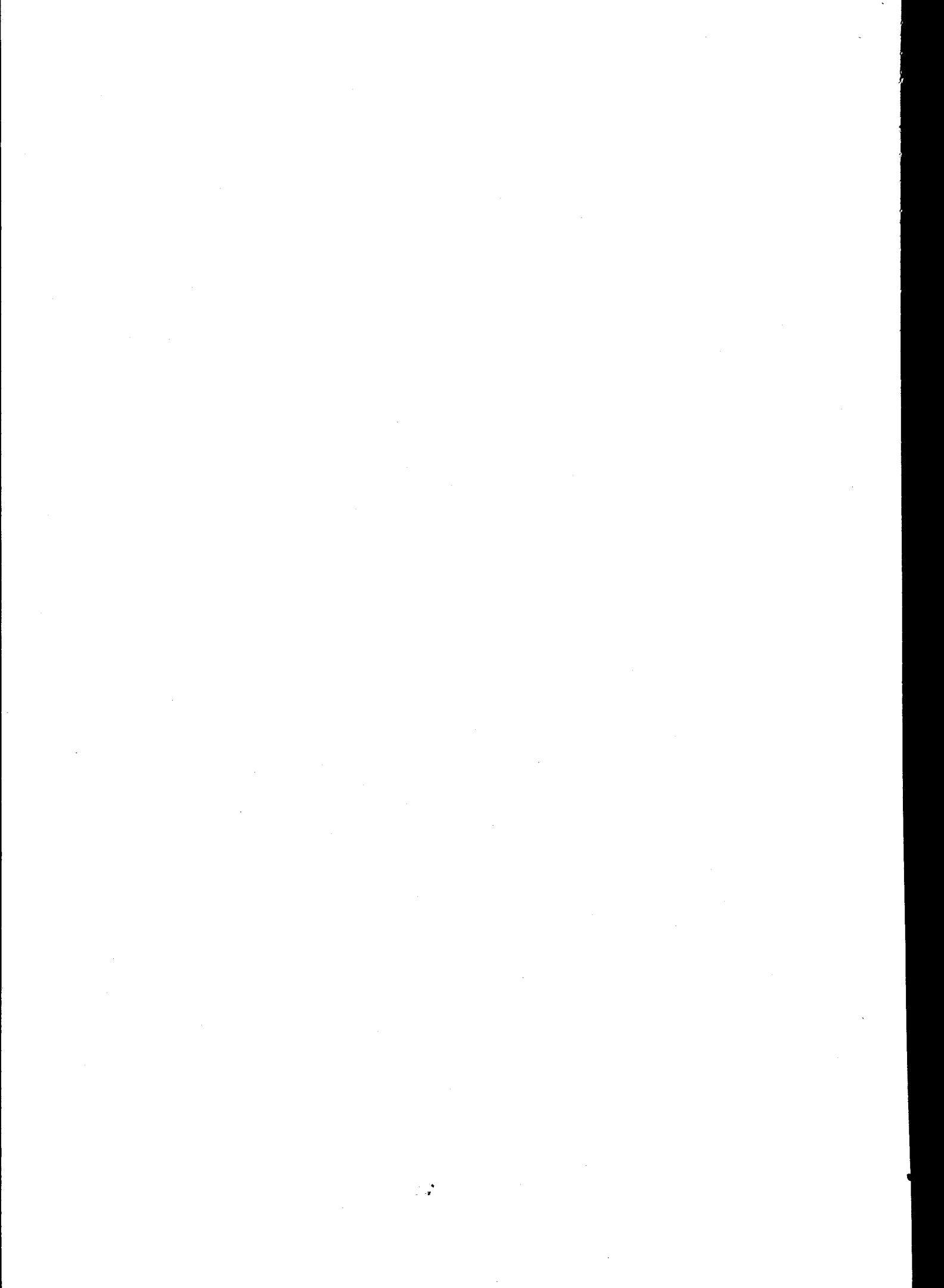
```
BEGIN  x := a; y := b;    { b ≥ 0 }
      z := 1;
      WHILE y > 0 DO
      BEGIN IF ODD(y) THEN z := z * x;
            y := y DIV 2;
            x := x * x
      END
      { z = ab }
END
```

(40/100)

(c) Pertimbangkan masalah mengemaskinikan dua fail yang diselenggarakan oleh sebuah universiti: fail *PELAJAR* mengandungi maklumat-maklumat pelajar yang sedang didaftarkan dengan universiti; fail *KELUAR2* mengandungi maklumat-maklumat pelajar yang telah dikeluarkan dari universiti tanpa memperolehi sebarang ijazah. Fail *KELUAR2* selalunya dirujuk, maka ianya diisi mengikut tertib menaik nombor pendaftaran. Operasi kemaskini dilakukan berasaskan kepada fail *IN* yang mengandungi nombor-nombor pendaftaran pelajar yang akan dikeluarkan dari universiti. Andaikan bahawa fail *IN* juga diisi mengikut tertib menaik nombor pendaftaran.

Lengkapkan program berikut supaya program yang dihasilkan akan mengemaskinikan fail *KELUAR2* (tanpa pengisihan) dan menghasilkan fail terisih baru, *PELAJARBARU* dan fail baru, *RALAT*. Fail *PELAJARBARU* mengandungi hanya pelajar-pelajar yang masih didaftarkan dengan universiti. Fail *RALAT* mengandungi nombor-nombor pendaftaran yang terdapat di dalam fail *IN* tetapi tidak di dalam fail *PELAJAR*.

Jenis-jenis rekod di dalam fail *PELAJAR*, *PELAJARBARU* dan *KELUAR2* tertakrif seperti berikut:



```

TYPE tarikh = RECORD tahun: integer;
                  bulan: (jan,feb,mac,apr,mei,jun,jul,
                        ogo,sep,okt,nov,dis);
                  hari: 1...31
            END;

rekelajar = RECORD nombor: integer;
                  nama: ARRAY[1...35] OF Char;
                  major: (mat,fiz,kom,kim,kaj);
                  tahun: (1,2,3,4)
            END;

keluar = RECORD rekod: rekelajar;
                  tarikhkeluar: tarikh
            END

```

Pembolehubah-pembolehubah di dalam program ini diisytiharkan seperti berikut:

```

VAR PELAJAR, PELAJARBARU : FILE OF rekelajar;
    KELUAR2 : FILE OF keluar;
    IN, RALAT : FILE OF integer

```

Andaikan bahawa '*hariini*' adalah suatu pembolehubah sejagat yang mengandungi tarikh semasa.

```

BEGIN
    pengawalan fail-fail;

    WHILE NOT EOP(IN) DO
        BEGIN WHILE IN↑ > PELAJAR↑.nombor DO
            kemaskinikan fail PELAJARBARU;

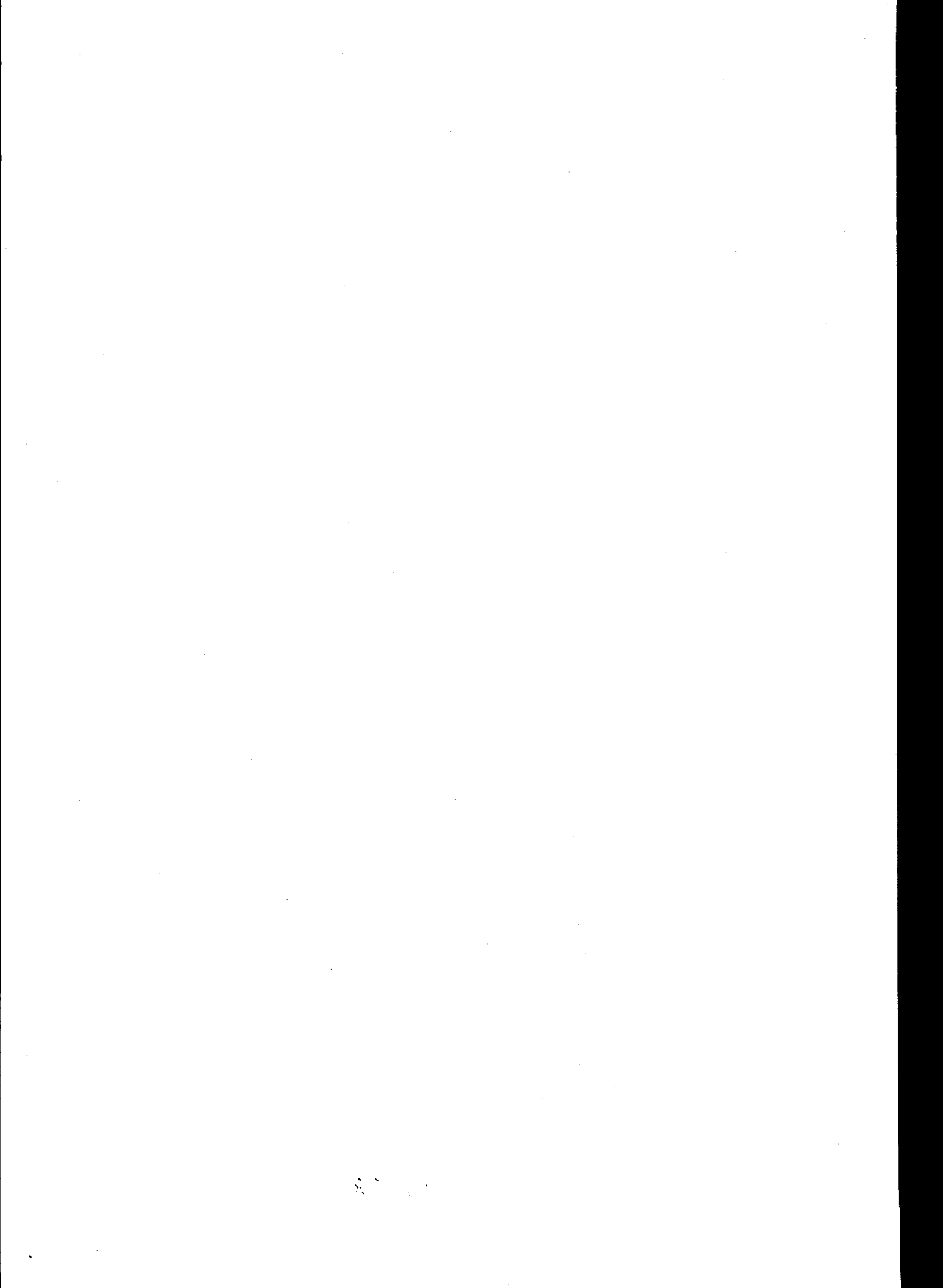
            IF IN↑ < PELAJAR↑.nombor THEN
                kemaskinikan fail RALAT

            ELSE kemaskinikan fail KELUAR2;

            GET(IN)

        END
    END
END

```



Untuk melengkapi program, anda perlu membangun setiap tembereng program yang ditunjukkan di atas, yaitu,

- (i) *pengawalan fail-fail*
- (ii) *kemaskinikan fail PELAJARBARU*
- (iii) *kemaskinikan fail RALAT*
- (iv) *kemaskintikan fail KELUAR2*

(40/100)

5. (a) Terangkan setiap yang berikut:

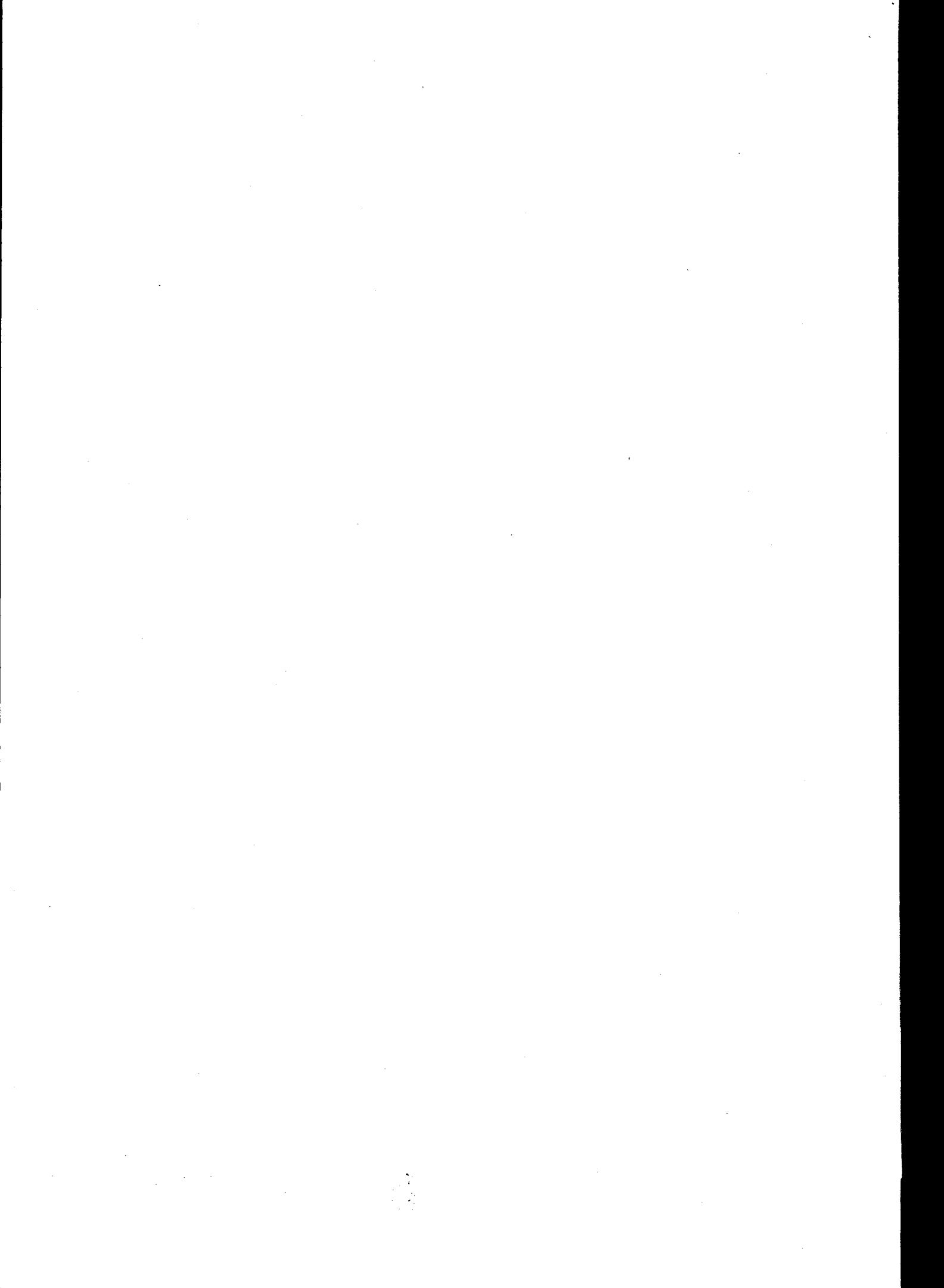
- (i) Fail teks
- (ii) Pengoperasi GET(f)
- (iii) Jenis skalar
- (iv) Struktur dinamik

(20/100)

(b) Buktikan bahawa program berikut tepat secara separa tetapi tidak tepat secara seluruh.

```
BEGIN { ( a > 0 ) ∧ ( b > 0 ) }  
    x := a; y := b;  
    { ( x > 0 ) ∧ ( y > 0 ) }  
    REPEAT WHILE x ≥ y DO x := x - y;  
        { 0 ≤ x < y }  
        WHILE y > x DO y := y - x  
        { 0 < y ≤ x }  
    UNTIL x = y  
END
```

(40/100)

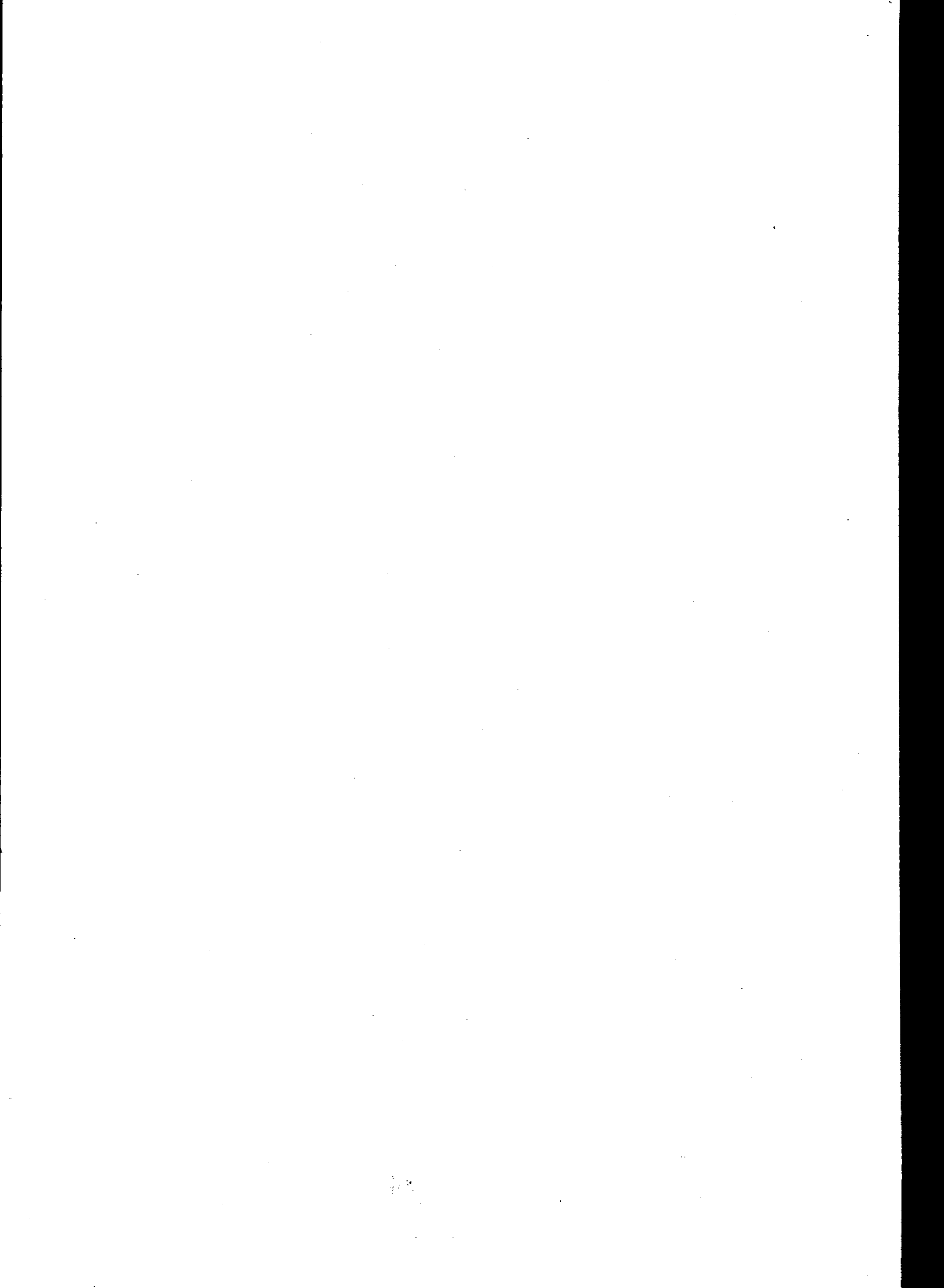


- (c) Tuliskan suatu program yang mencari nilai minimum, *MIN*, bagi unsur-unsur dari suatu tatasusunan satu dimensi *A*. Program ini juga mengumpulkan indeks unsur minimum itu kepada pembolehubah *M*.

Buktikan ketepatan separa bagi program yang telah dituliskan itu.

(40/100)

...ooOoo...



Petua-petua untuk ketepatan programLAMPIRAN

1. (a)
$$\frac{\{P\} S \{R\}, R \Rightarrow Q}{\{P\} S \{Q\}}$$
 (b)
$$\frac{P \Rightarrow R, \{R\} S \{Q\}}{\{P\} S \{Q\}}$$
2.
$$\{P_e\} x := e \{P\}$$
3.
$$\frac{\{P_{i-1}\} S_i \{P_i\} \text{ for } i = 1, \dots, n}{\{P_0\} \text{ begin } S_1; S_2; \dots; S_n \text{ end } \{P_n\}}$$
4. (a)
$$\frac{\{P \wedge B\} S_1 \{Q\}, \{P \wedge \neg B\} S_2 \{Q\}}{\{P\} \text{ if } B \text{ then } S_1 \text{ else } S_2 \{Q\}}$$
 (b)
$$\frac{\{P \wedge B\} S \{Q\}, P \wedge \neg B \Rightarrow Q}{\{P\} \text{ if } B \text{ then } S \{Q\}}$$
5.
$$\frac{\{P \wedge (x = k_i)\} S_i \{Q\} \text{ for } i = 1, \dots, n}{\{P \wedge (x \in \{k_1, \dots, k_n\})\} \text{ case } x \text{ of } k_1 : S_1; \dots; k_n : S_n \text{ end } \{Q\}}$$
6.
$$\frac{\{P \wedge B\} S \{P\}}{\{P\} \text{ while } B \text{ do } S \{P \wedge \neg B\}}$$
7.
$$\frac{\{P\} S \{Q\}, Q \wedge \neg B \Rightarrow P}{\{P\} \text{ repeat } S \text{ until } B \{Q \wedge B\}}$$
8. (a)
$$\frac{\{(a \leq x \leq b) \wedge P([a \dots x])\} S \{P([a \dots x])\}}{\{P([\])\} \text{ for } x := a \text{ to } b \text{ do } S \{P([a \dots b])\}}$$
 (b)
$$\frac{\{(a \leq x \leq b) \wedge P([x \dots b])\} S \{P([x \dots b])\}}{\{P([\])\} \text{ for } x := b \text{ downto } a \text{ do } S \{P([a \dots b])\}}$$

